MSCI 446 – Fall 2016

Tut #2: Exploratory Data Analysis

Shivangi Chopra

UNIVERSITY OF WATERLOO

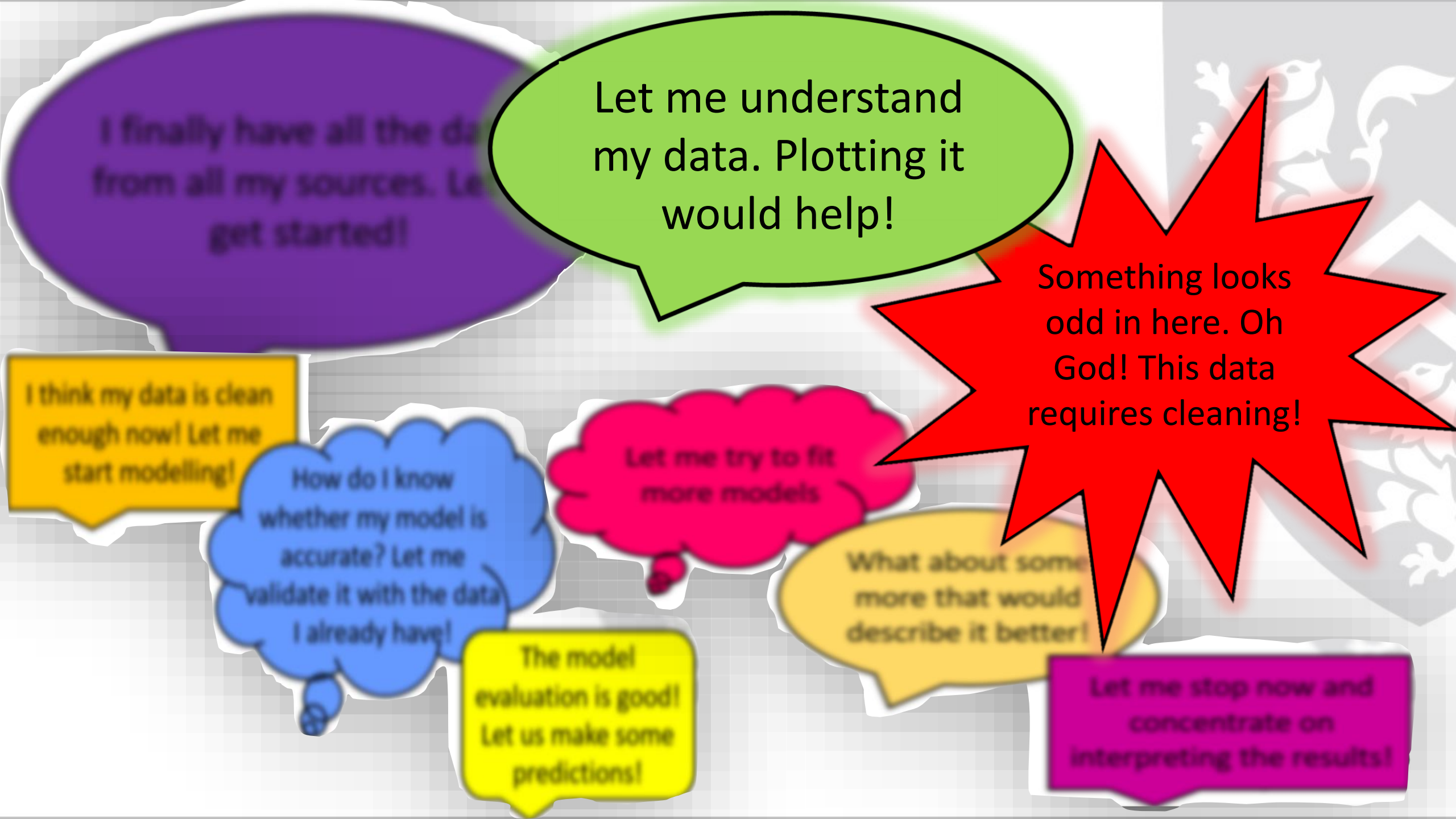
Recap of Tutorial 1

- Tools used for Data Science
- How to install Python and the required packages
  - Basics of Python
  - Anaconda
  - pip
  - Jupyter Notebooks
  - Pycharm IDE
  - Import csvs, import libraries, make scatter plots
- Introduction to WEKA – colour-coded histograms, scatter plots
- Introduction to Orange – import and view csvs, make scatter plots

Objectives

- Why is graphing important?
- Kinds of graphs for
  - Categorical data
  - Numerical Univariate data
  - Numerical Bivariate data
  - Numerical and Categorical data
  - Numerical Multivariate data
  - Other Kinds of data (text, images)

UNIVERSITY OF WATERLOO


I finally have all the da
from all my sources. Le
get started!
Let me understand my data. Plotting it would help!
Something looks odd in here. Oh God! This data requires cleaning!
I think my data is clean enough now! Let me start modelling!
How do I know whether my model is accurate? Let me validate it with the data I already have!
Let me try to fit more models
What about some
more that would
describe it better!
The model evaluation is good! Let us make some predictions!
Let me stop now and concentrate on interpreting the results!

Motivation

- We will look at the anscombe_quartet
- It consists of 4 datasets with 11 bivariate (x,y) points in each

Anscombe_i

	x	y
0	10	8.04
1	8	6.95
2	13	7.58
3	9	8.81
4	11	8.33
5	14	9.96
6	6	7.24
7	4	4.26
8	12	10.84
9	7	4.82
10	5	5.68

- We will calculate the summary statistics (namely mean and stdev) of the x and y column of each of the datasets
- Notice that the mean and stdev of the 4 datasets are the same
- What does that tell us about the 4 datasets?

Data Set I

	x	y
mean	9.000000	7.500909
std	3.316625	2.031568

Data Set II

	x	y
mean	9.000000	7.500909
std	3.316625	2.031657

Data Set III

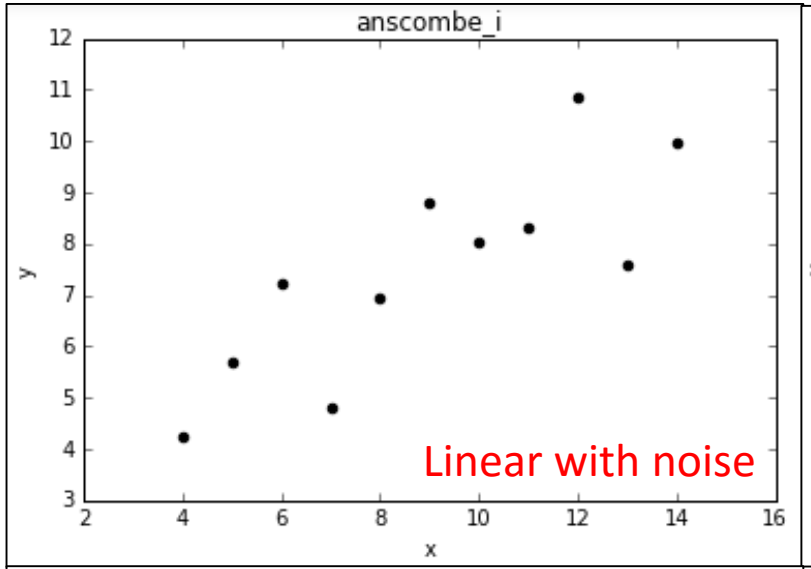
	x	y
mean	9.000000	7.500000
std	3.316625	2.030424

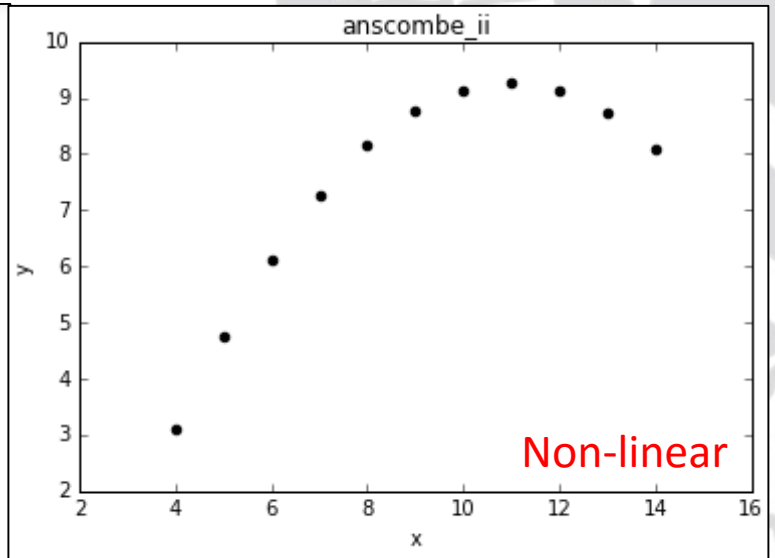
Data Set IV

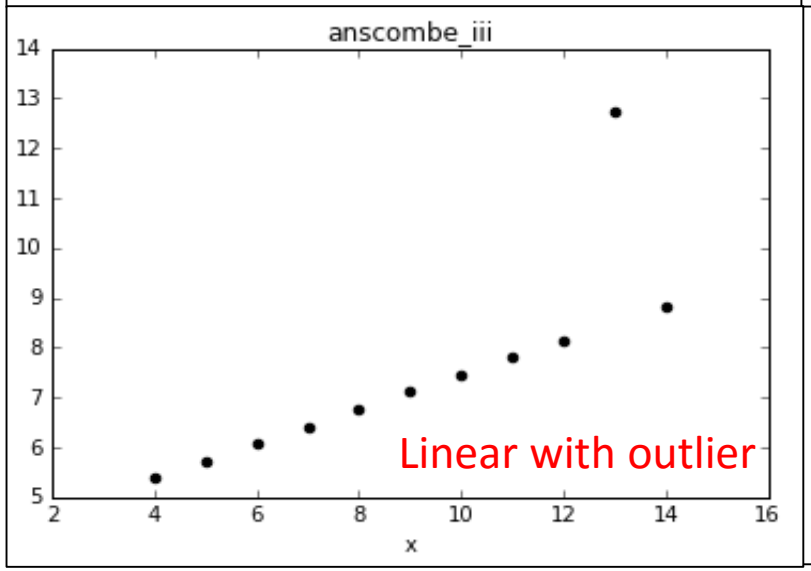
	x	y
mean	9.000000	7.500909
std	3.316625	2.030579

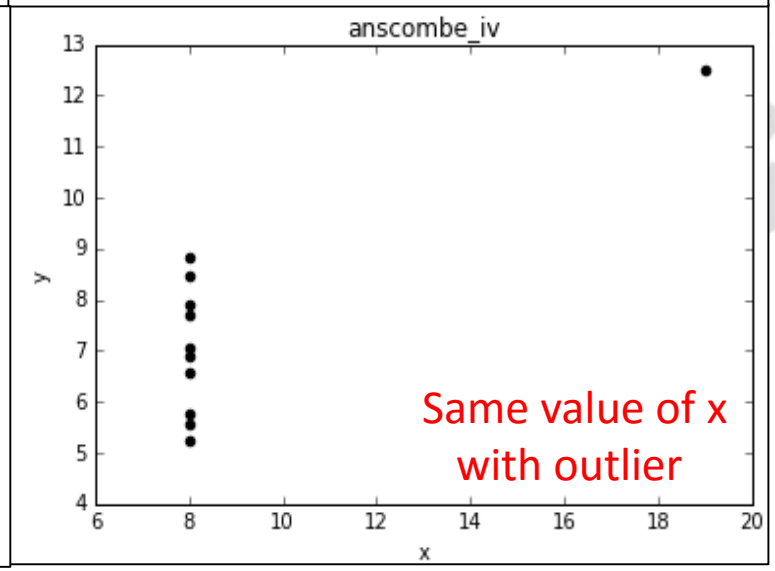
Let us plot the anscombe_quartet

- Even though the summary statistics hinted that the 4 datasets might have the same trends, creating scatter plots for each reveals that this is not the case
- Therefore, it is very important to plot the data and see how variables correlate with each other before any modelling



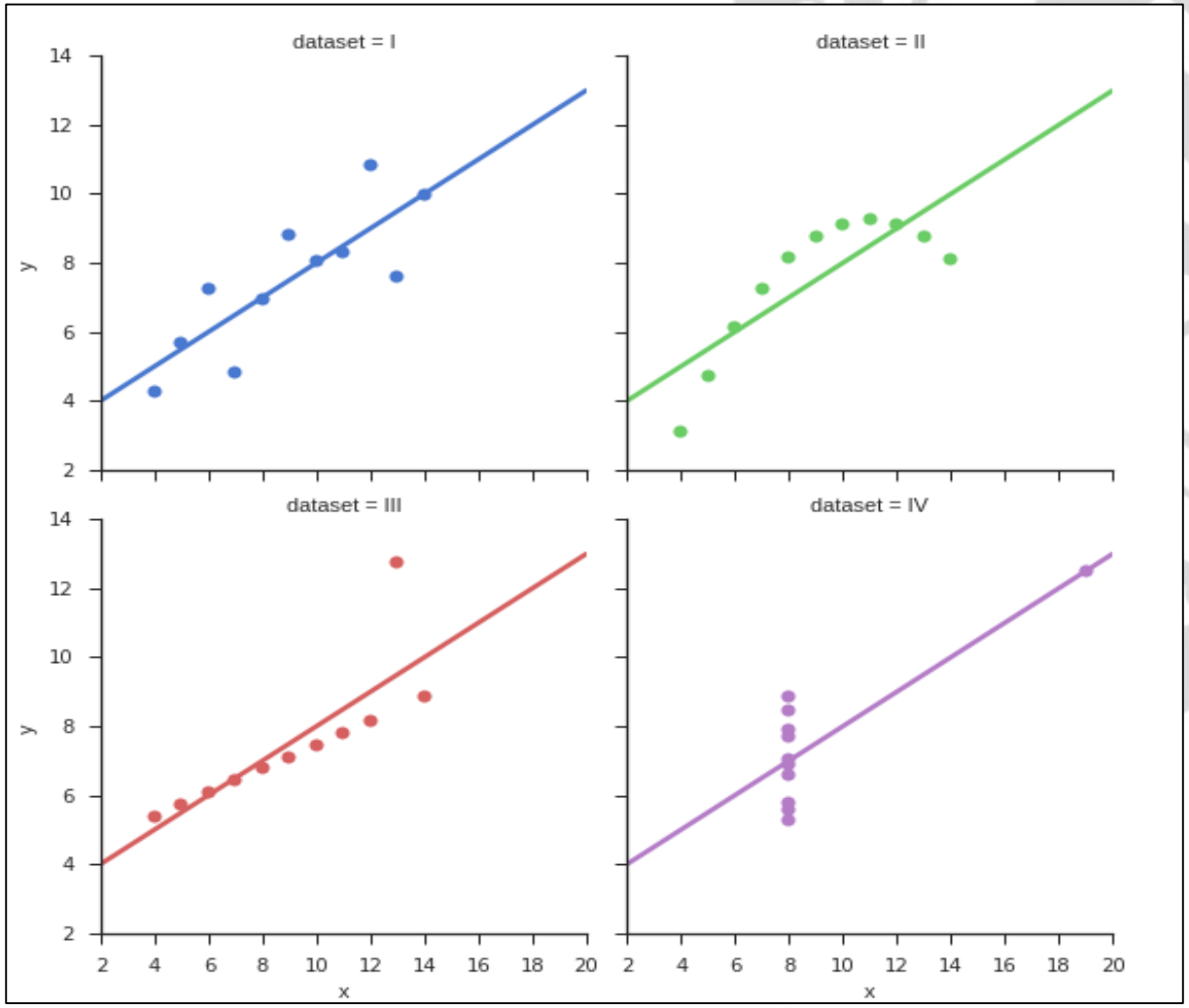






Python code for the previous example

- The datasets can be found on learn under the datasets folder
- The code can be found in the 446_2_motivation.txt file
- If running in Jupyter notebooks, add *%matplotlib inline* after the import statements to show the graphs as inline outputs
- If running in PyCharm, add *plt.show()* after each plot to show the plots during the output



Categorical Variables

- Gender, designation..
- A frequency table (count of each category) is a common statistic used for describing categorical data
- Pie and bar charts are used to show the above
- For all further demonstration, we will use information on academicians from the salary2015.xlsx available in the datasets folder in Learn
- Code is available in 446_2_categorical.txt

Categorical Variables Example

```
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns

salaryall = pd.read_excel('salary2015.xlsx')
print("Total number of employees at UW: ", len(salaryall))

#getting rows of only academicians
salary_academics = salaryall[salaryall.position.isin(
    ['Professor','Lecturer','Associate Professor','Assistant Professor'])]

print("Total number of academicians at UW: ", len(salary_academics))
print(salary_academics[0:10])

#get unique values of "position" and their counts
count = salary_academics.position.value_counts()
print(count)
```

```
Total number of employees at UW:  1295
Total number of academicians at UW:  1010
```

	position	salary_paid	taxable_benefits
0	Associate Professor	138511.04	450.56
1	Associate Professor	151941.80	252.04
2	Assistant Professor	135039.16	207.36
3	Assistant Professor	127011.48	491.80
4	Professor	173552.68	606.08
5	Associate Professor	128160.36	535.76
6	Associate Professor	109110.60	301.68
7	Professor	178136.32	290.56
8	Associate Professor	159481.92	264.48
9	Lecturer	123897.44	187.28

```
Professor                 420
Associate Professor       388
Assistant Professor       115
Lecturer                   87
```

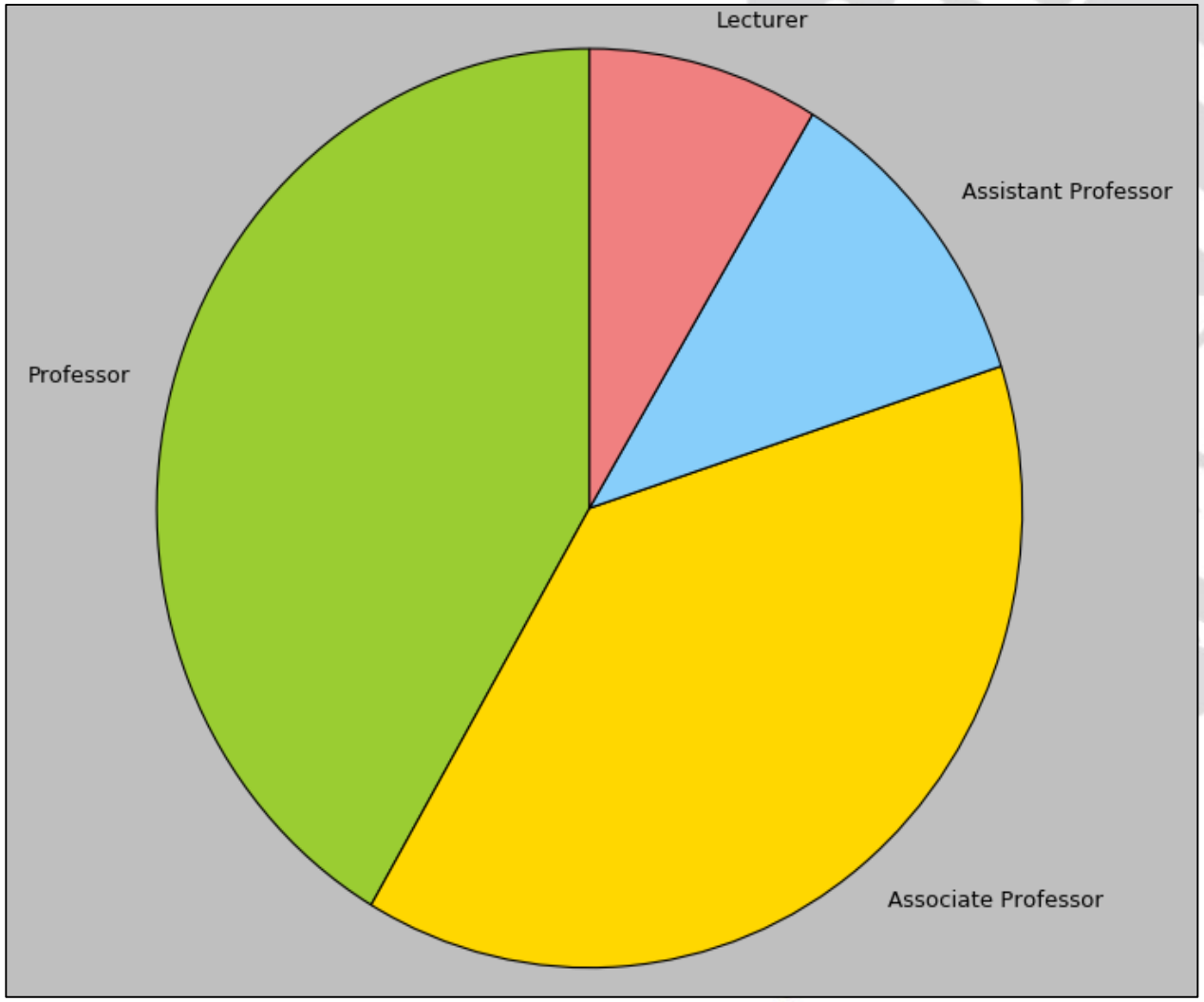
UNIVERSITY OF WATERLOO

Pie Chart

```
#get unique values of "position" and their counts
count = salary_academics.position.value_counts()
print(count)

# PIE CHART
# The slices will be ordered and plotted counter-clockwise.
colors = ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral']
labels = count.index.values
values = count.values

plt.pie(values, labels=labels, colors=colors, startangle=90)
# Set aspect ratio to be equal so that pie is drawn as a circle.
plt.axis('equal')
plt.show()
```

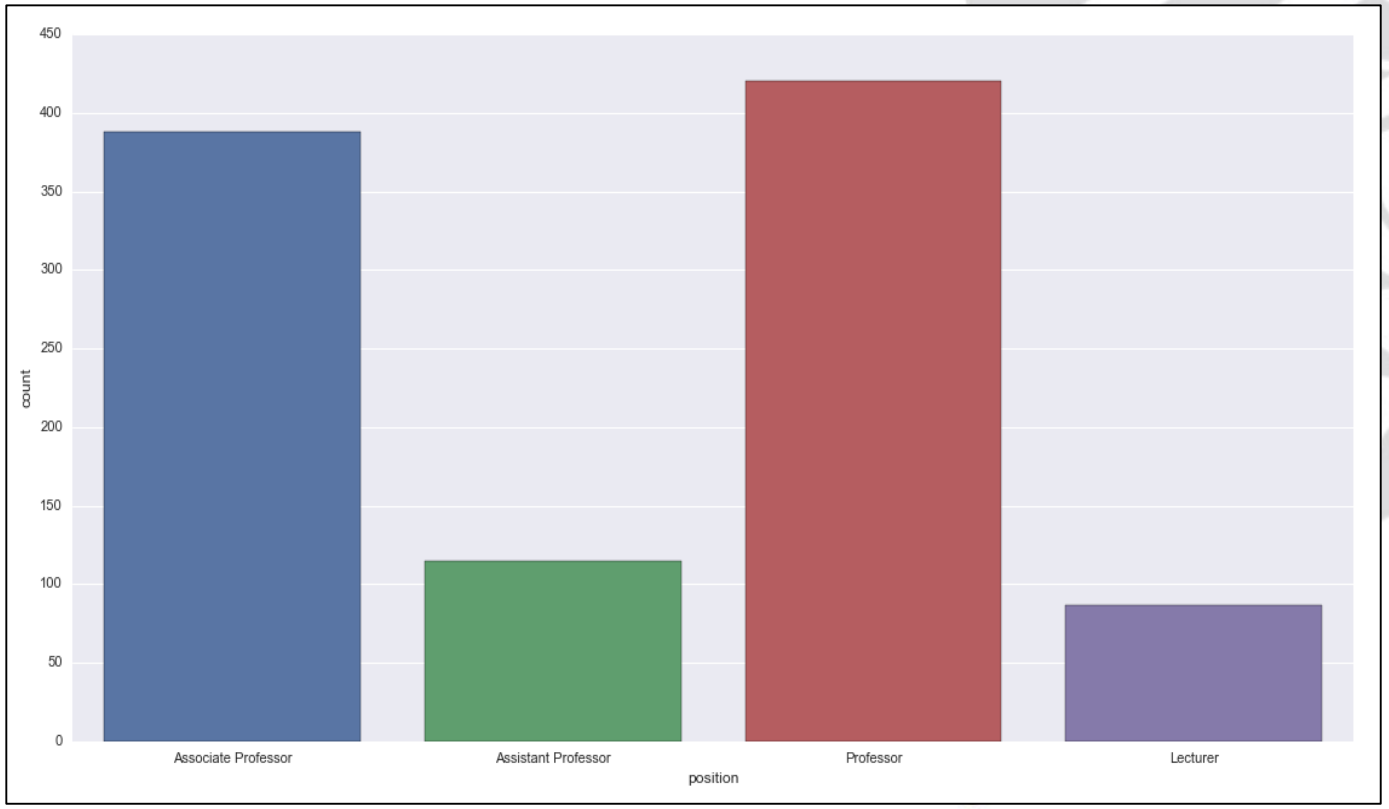


- Are there more Associate Professors or Professors?
- Are there more Lecturers or Assistant professors?

Bar Chart

```
# BAR CHART
sns.countplot(salary_academics.position, data=salary_academics)
plt.show()
```

- Are there more Associate Professors or Professors?
- Are there more Lecturers or Assistant professors?
- It is difficult for humans to understand angles of the pie chart. Bar charts are easier to understand.

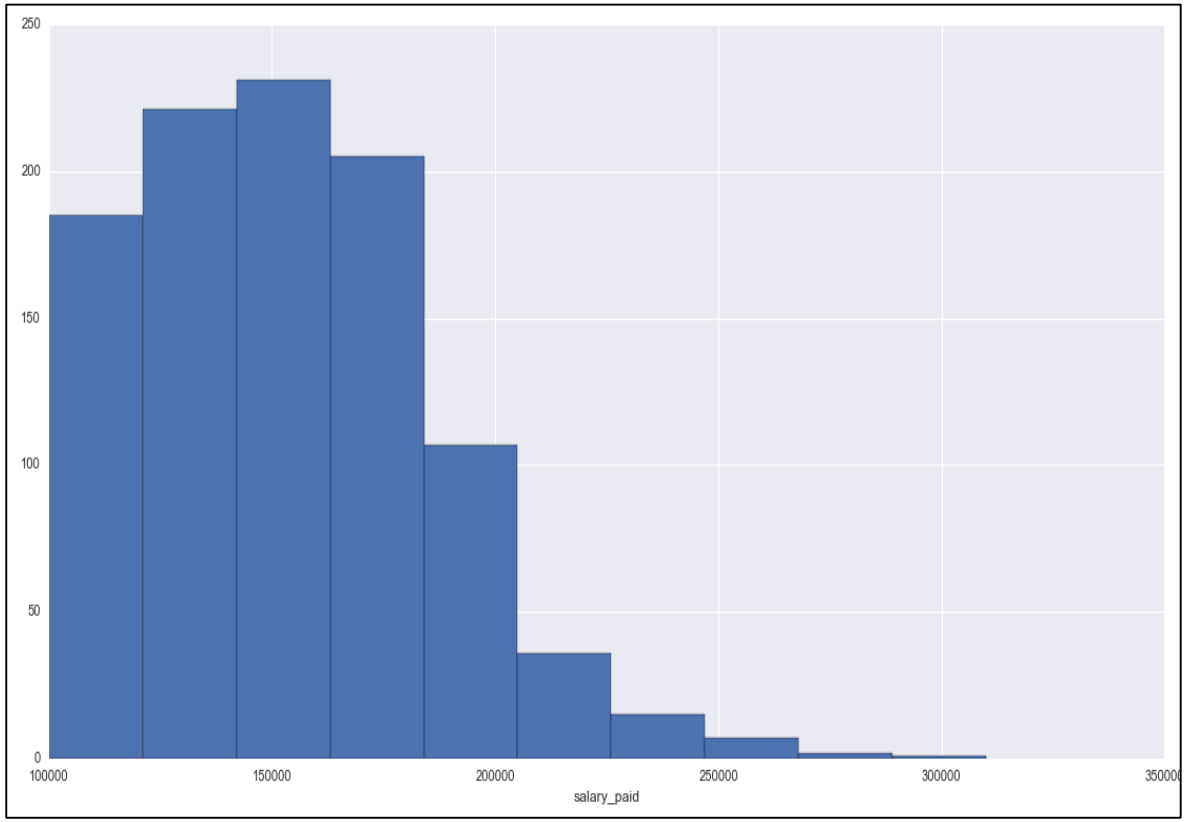


Numerical Variables – Univariate

- Weight, Height, Marks, Salary..
- Want to know the characteristics of a single variable; its frequency distribution, mean, median, percentiles etc.
- Histograms and box and whisker plots are used to show the above
- The salary paid to the academicians of UW in 2015 is analyzed using the above
- Code can be found on learn in 446_2_univariate.txt

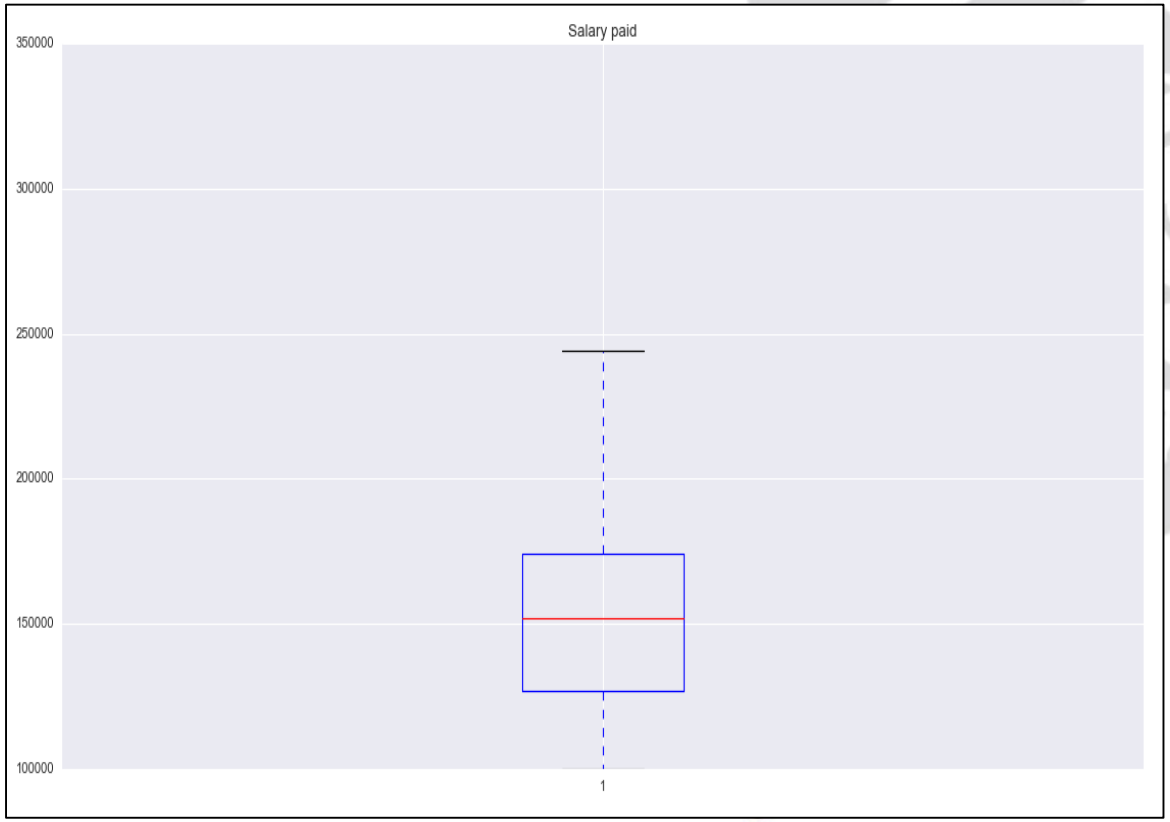
Histogram

```
plt.figure(1)
plt.hist(salary_academics["salary_paid"])
plt.xlabel("salary_paid")
plt.show()
```



Box and whisker plot

```
plt.figure(2)
plt.boxplot(salary_academics.salary_paid)
plt.title("Salary paid")
plt.show()
```

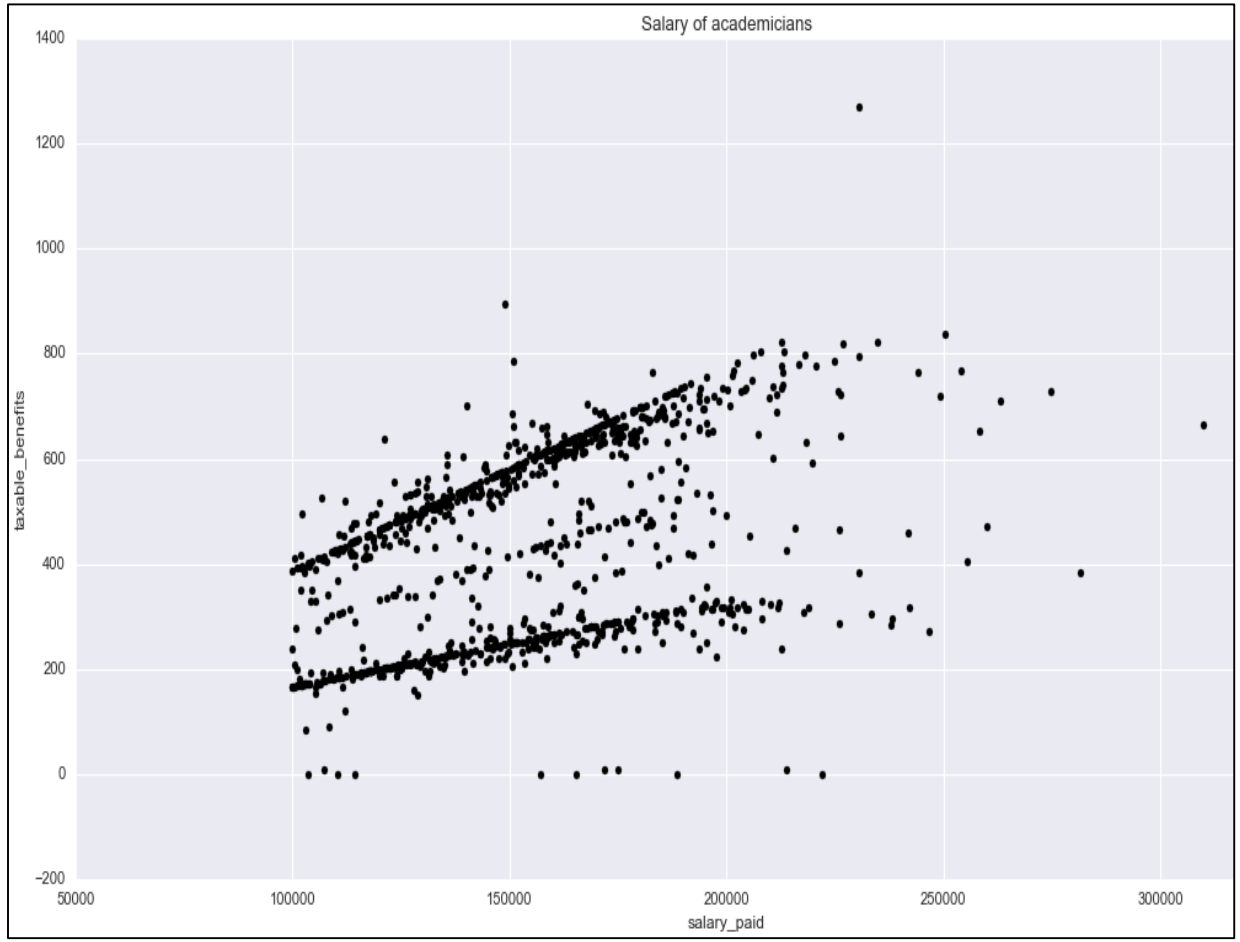


Numerical Variables – Bivariate

- Need to understand the interaction between 2 variables and decipher their correlation. A scatter plot comes in handy here
- A scatter plot of salary vs. taxable benefits for all 1996 employees of UW was plotted in the last tutorial
- If color can be added to this graph based on a third variable's value (categorical), it can help us understand how the 2 variables interact w.r.t the 3rd one as well
- This is done by plotting salary vs. taxable benefits for the UW academicians of 2015. A color is attributed to each point based on their designation
- Code can be found on learn in 446_2_bivariate.txt

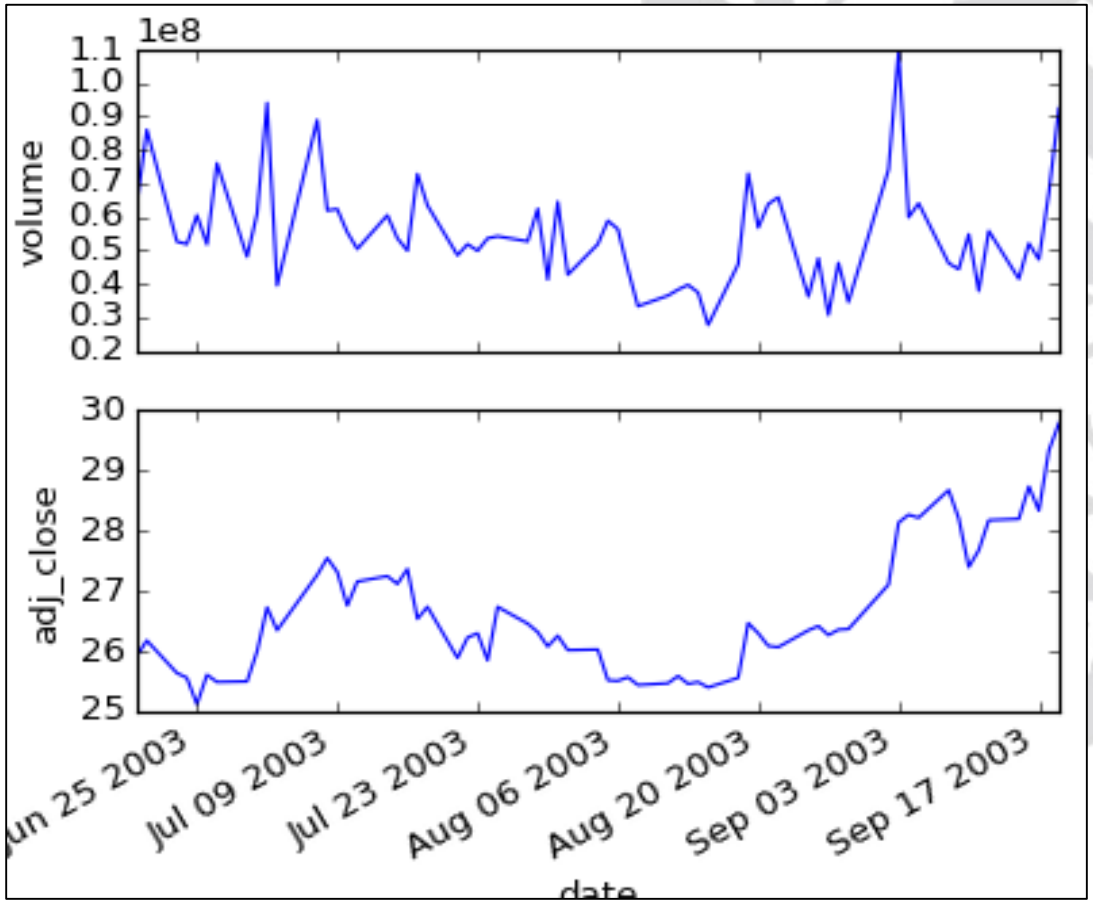
```
plt.figure(1)
plt.scatter(salary_academics.salary_paid, salary_academics.taxable_benefits,  color='black')
plt.title("Salary of academicians")
plt.xlabel("salary_paid")
plt.ylabel("taxable_benefits")
```

```
# scatter plot with color based on a categorical variable
# create a grid first and then map the graph to this grid
position = ['Professor','Lecturer','Associate Professor','Assistant Professor']
fg = sns.FacetGrid(data=salary_academics, hue='position', hue_order=position)
fg.map(plt.scatter, 'salary_paid', 'taxable_benefits').add_legend()
```



Numerical Variables – Bivariate

- Apart from scatter plots, there are other plots that can be used to visualize bivariate data
- Time series can be shown using line chart (Code:http://matplotlib.org/examples/pylab_examples/plotfile_demo.html)
- For other kinds of charts –
  - http://matplotlib.org/gallery.html
  - https://stanford.edu/~mwaskom/software/seaborn/examples/

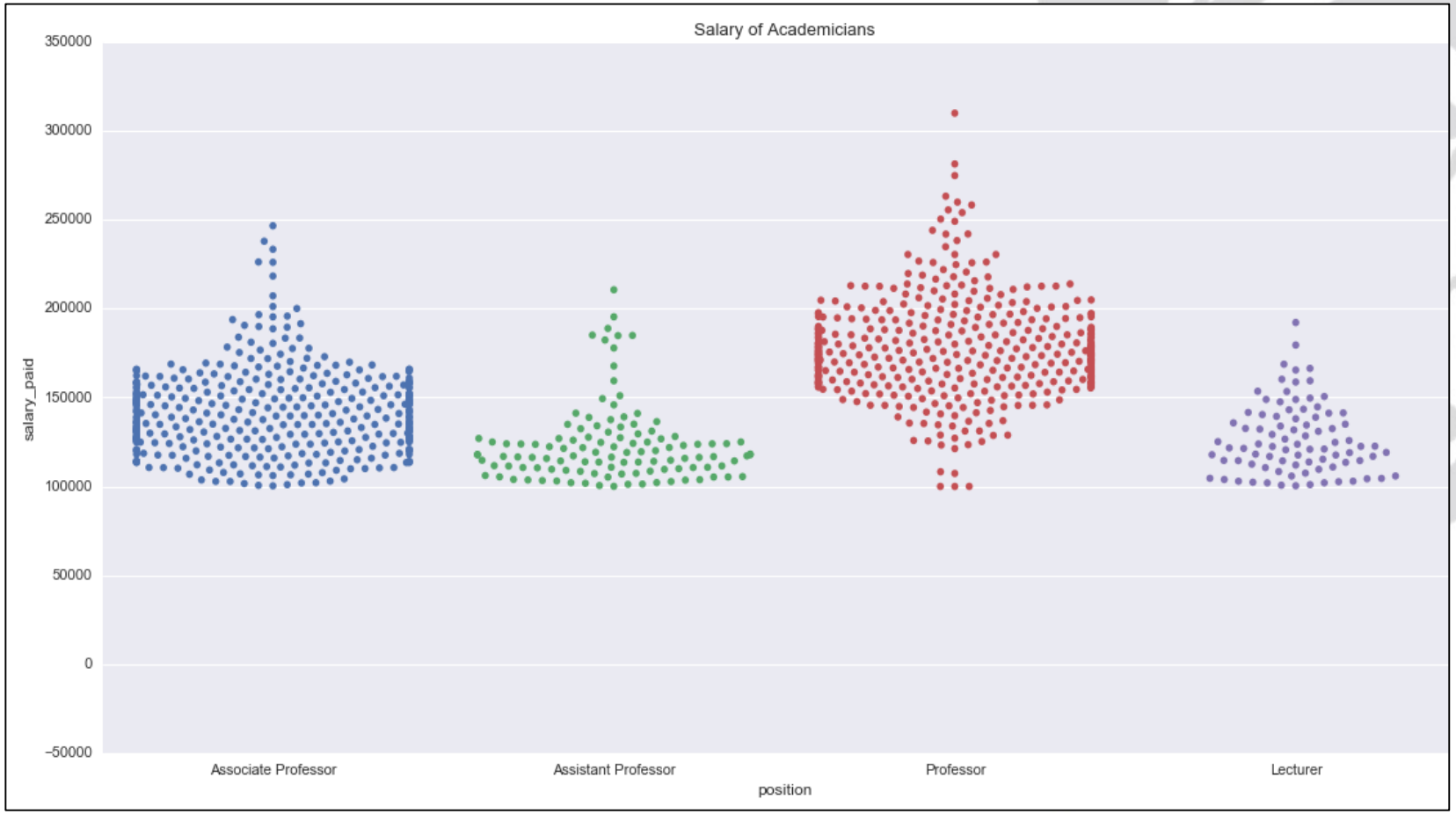


UNIVERSITY OF WATERLOO

Numerical and Categorical Variables – Bivariate

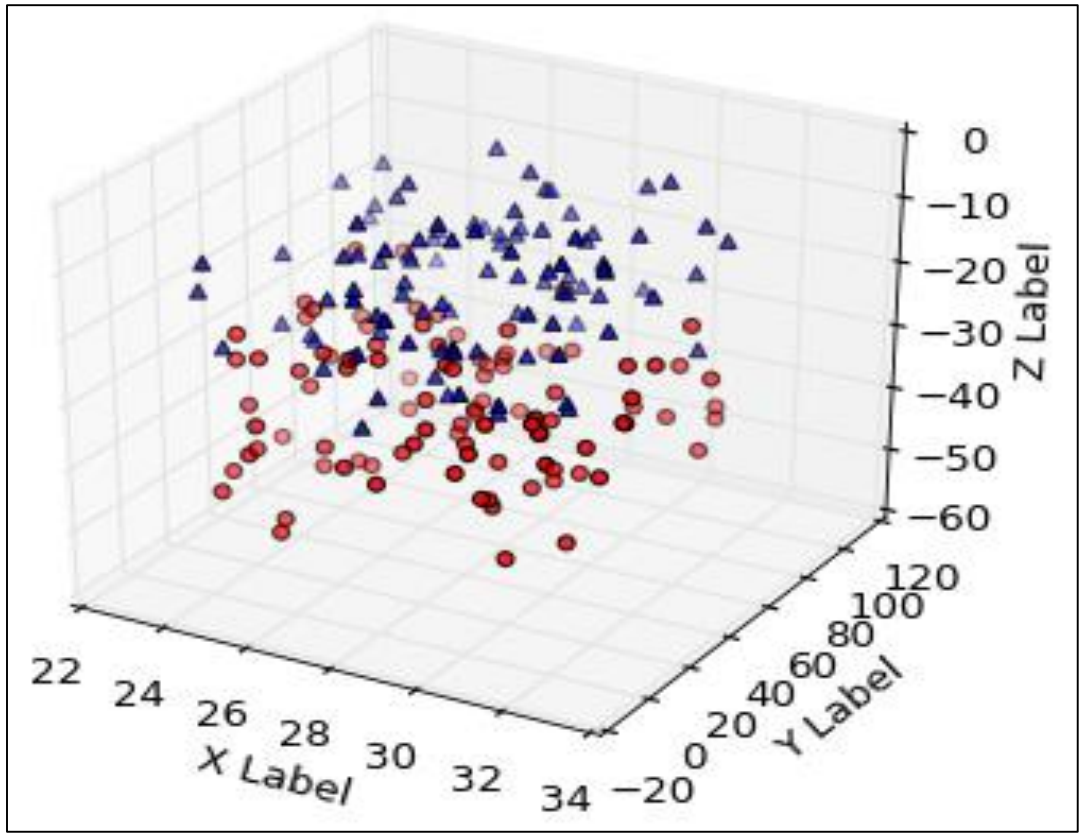
- If we want to see how many employees of each category get what salary, a swarm plot is good visualization
- We can also compare the ranges of salaries of the different categories of the employees

```
sns.swarmplot(x=salary_academics.position, y=salary_academics.salary_paid, data=salary_academics)
```

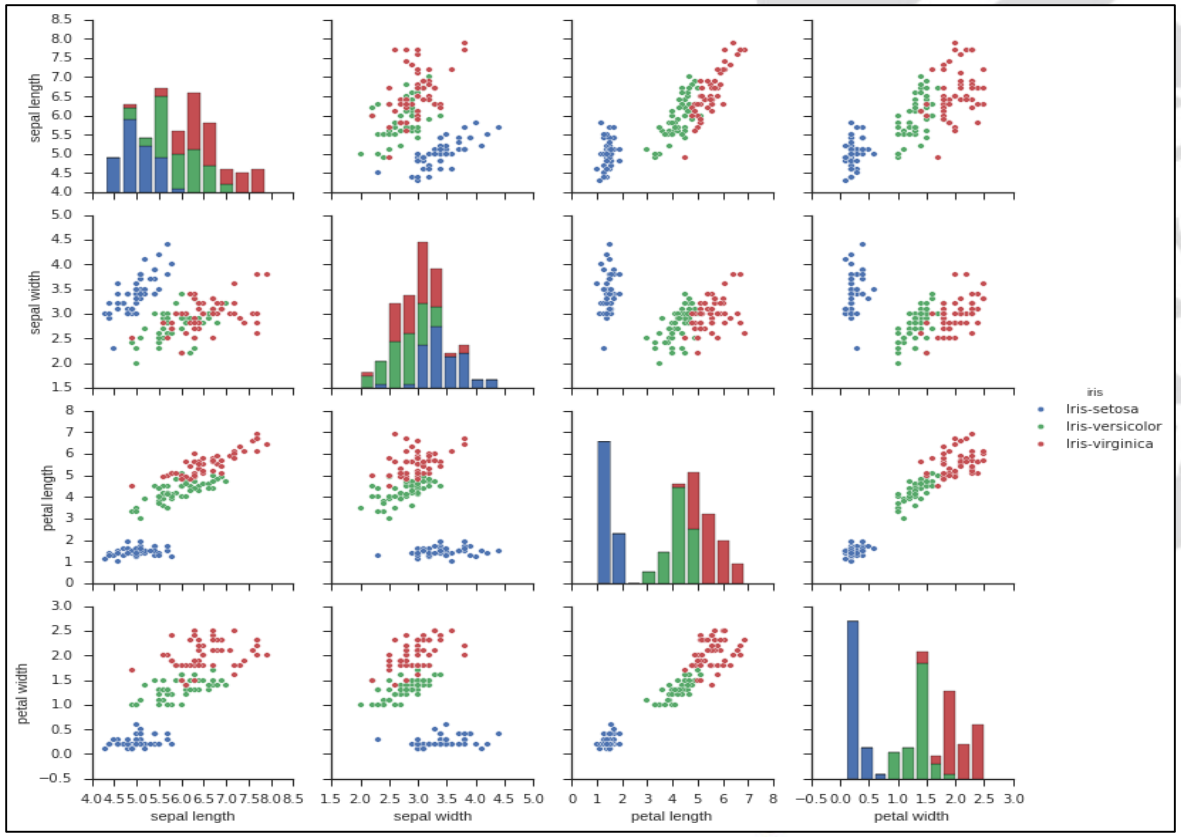


Numerical Variables - Multivariate

Available
(http://matplotlib.org/examples/mplot3d/scatter3d_demo.html)

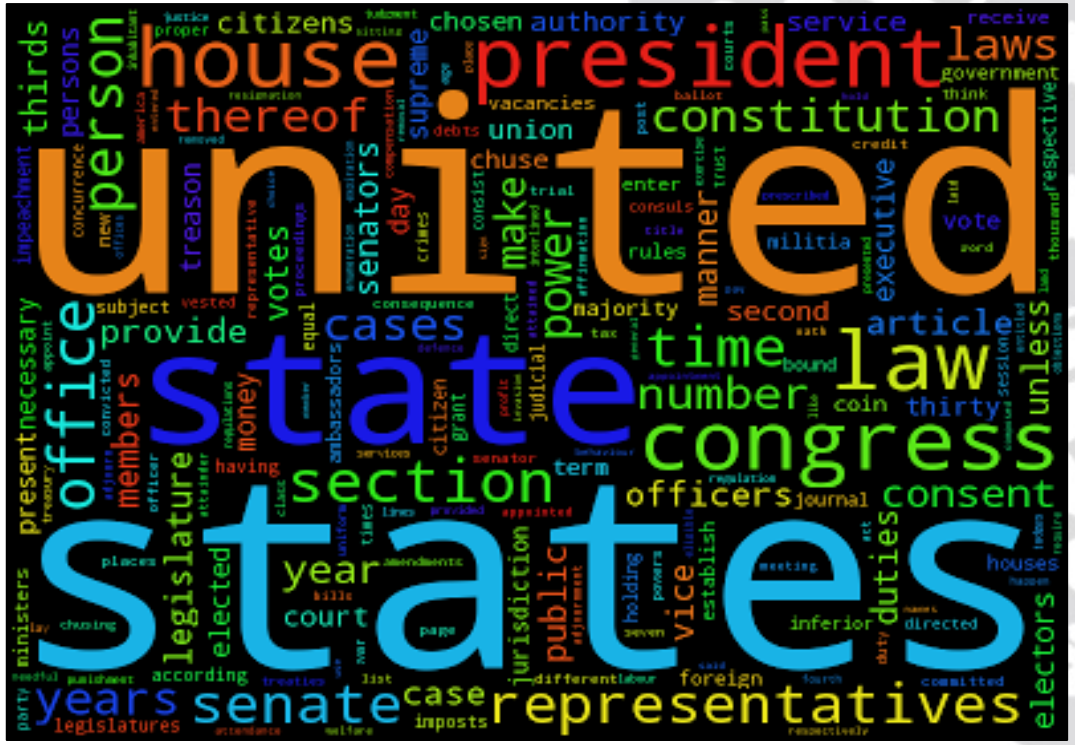


Recommended (easy to understand)
(*sns.pairplot(entire_dataset, hue = categorical_feature))*



Natural Language Text variables

- Given a paragraph/article or any natural language text, the main words/descriptors can be found considering its frequency in the article vs. its frequency in the English Language
- Word clouds are graphical representations of the common topics of the article and the sizes of the words represent their importance in the article
- Sentiment analysis can be run to categorize opinions as positive/negative/neutral



word cloud (frequently occurring words) on the US constitution (https://github.com/amueller/word_cloud)

UNIVERSITY OF WATERLOO

Recap of what we did today…

- Summary statistics can be deceptive
- Kinds of graphs for
  - Categorical data – pie chart (bad), bar chart
  - Numerical Univariate data – histogram, box and whisker plot
  - Numerical Bivariate data – scatter plot, line charts
  - Numerical and categorical data – swarm plots, violin plots
  - Numerical Multivariate data – plot all variables against another
  - Other kinds of data (text, images) – word clouds, decomposed images